DOCUMENT RESUME

ED 084 843 EM 011 664

AUTHOR Halley, Fred S.
TITLE SOLIB: A Social Science Program Library for Small Computers.
INSTITUTION State Univ. of New York, Brockport. Coll. at Brockport.
PUB DATE Jun 73
NOTE 4p.; Paper presented at the Conference on Computers in the Undergraduate Curricula (Claremont, California, June 18 through 20, 1973)

EDRS PRICE MF-$0.65 HC-$3.29
DESCRIPTORS *Computer Assisted Instruction; Computer Oriented Programs; *Computer Programs; *Data Analysis; *Data Processing; Higher Education; Program Descriptions; *Sociology
IDENTIFIERS FORTRAN; IBM 1130 Computer; *Sociology Library; SOLIB; Transportability

ABSTRACT

A package of social science programs--Sociology Library (SOLIB)--for small computers provides users with a partial solution to the problems stemming from the heterogeneity of social science applications programs. SOLIB offers a uniform approach to data handling and program documentation; all its programs are written in standard FORTRAN for the IBM 1130 computer and designed to run in 16k words. SOLIB programs can be divided into three groups, dealing with data preparation, data organization and data analysis. The three programs of the first category are used to clean data, check missing or duplicate cards and to clean wild codes; two data organization programs conduct item analyses and perform generalized recoding, allowing the user to rearrange his data without programing knowledge. A pair of data analysis programs creates frequency distributions and performs cross-tabulations. The system is being used successfully by students in connection with course work and independent study and by faculty for professional research. Although transportability of SOLIB to other campuses and different computers requires some program modifications, these are minimal and conversion can be accomplished with little difficulty. (LB)

FILMED FROM BEST AVAILABLE COPY

ED 084843

SOLIB: A SOCIAL SCIENCE PROGRAM LIBRARY FOR SMALL COMPUTERS

Fred S. Halley
Department of Sociology
Brockport College, State University of New York
Brockport, New York 14420
(716) 395-2618

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

The intent of this paper is to describe a package of social science programs for smaller computers and its use and acceptance by students and faculty.

The absence of social science programming packages appropriate for small computing systems is particularly noticeable to faculty members who have trained in major universities where program packages such as the Bicmedical, Omnitab, Statistical Programs for Social Science, or other similar packages are available at large computing installations. More often than not, computer centers at smaller schools try to meet social scientists' programming needs on an ad hoc basis, the result being a potpourri of locally written programs or modifications of programs which have been written elsewhere. Such a computer center library presents students and less sophisticated faculty users with a bewildering variety of approaches to data handling and documentation which hinders the teaching of computer applications for research in the social sciences.

SOLIB (SOciology LIBrary) is a partial solution to the problems of the heterogeneity found in social science applications programs. Like the major social science packages, SOLIB provides a uniform approach to data handling and program documentation. In a uniform manner, it provides a series of programs which may be used to edit and remove wild codes from data, to organize data for analysis, and to perform initial data analysis and description. To facilitate use of SOLIB, a thirty-two page users' manual has been published which contains guidelines for usage in addition to individual program documentation.

The SOLIB Programs

Although each of the SOLIB programs was designed to aid the user in different aspects of the research process, an attempt was made to create a uniform approach to the control cards necessary to specify information for each unique application. The cards used to call specific programs, select options, identify and label variables, and end card files are the same for all the programs. In this manner, the general logic of use is consistent from program to program minimizing operational difficulties for the user.

The programs have been written in standard FORTRAN for the IBM 1130 computer. While this FORTRAN has some of the features of the full FORTRAN IV, it is more similar to FORTRAN II. The programs may be easily converted to other machines with FORTRAN compilers. The package in use at Brockport runs in 16K words; a second version has been developed which runs in 8K words and is in operation at Augustana College in Sioux Falls, South Dakota.

Two features not available on most smaller computers are the END parameter in the FORTRAN read statement and the variable format option used to set formats at execution time. The absence of these features has made it necessary for most 1130 programs to have fixed formats, or to require the user to write a data input subroutine. The absence of an END parameter has required the user to know the exact number of cards in his data deck. While these inconveniences do not pose obstacles to the experienced user, they create problems for the novice.

To remedy these problems, the SOLIB programs use a common set of data input routines which read all data cards in character format. Conversion routines then change character data in user-specified card columns to numeric values. Variables are specified by variable identification cards instead of the usual FORTRAN format statement. For each variable, the user supplies a variable identification card which specifies the columns in which the variable appears on his data cards and an alphanumeric label for the variable. This identification procedure is more easily understood by the novice user than the FORTRAN format statement since it refers to variables by column numbers. By placing the alphanumeric label information on the same variable identification card with the column numbers, labeling and formating confusions are avoided. While this procedure requires that a variable identification card be punched for each variable, the variable identification cards may be used with all the SOLIB programs, and the variable identification lists may be easily modified by the insertion or deletion of variable identification cards.

ED 084843

Since the entire data card is read in character format, it is possible to test for special characters as the cards are read. In this manner, a card with a $ in column 1 has been used as an "end of file" card for all the programs. By being able to detect the end of a data file, it is possible to make the programs self-counting, that is, it is unnecessary for the user to know the number of cards in his deck. When the $ card is detected, reading stops and computations progress.

The programs can be divided into three groups: data preparation, data organization and data analysis.

There are three data preparation programs. The first two aid the researcher in "cleaning" his data. Most data, regardless of how carefully it is punched, contains errors. Cases are punched twice or skipped entirely, and out-of-range codes are inadvertently punched. The data cleaning programs may be enlisted to help find these common errors.

The first data cleaning program checks data deck sample number to insure that there are no missing or duplicate cards. Sequential case numbering is assumed. If the program detects duplicate or missing cases, their presence or absence is reported on the line printer. The second data cleaning program performs wild code checking. The user specifies the maximum value each variable should attain. The program scans each data card. If a value greater than the maximum specified is located, the case number, the column numbers, the error and the variable name are reported. Each card is also checked for non-numeric characters. If a non-numeric character is encountered, an error message is printed along with the entire card image. In addition to these two data cleaning programs, a card listing program is also provided so that card decks may be listed on the line printer for further visual checking and backup purposes in case a card is lost or damaged.

SOLIB has two programs designed for data organization. The first is for item analysis to help the user summarize data by constructing indices. The program computes point biserial correlations between items and total item scores. The total scores are computed for each case by the program; for missing data, the mean for the items is substituted. After the point biserial correlations have been computed and printed, the items with the lowest correlations are eliminated from the item list and the analysis is repeated. The number of items eliminated each round and the number of rounds is set by the user.

The second data organization program is a generalized recoding program which allows up to nine types of recoding and modification of data decks. The program reads up to five old decks and produces a new deck through the use of nine recoding and modification options. The columns from which information on the original decks will be obtained and placed on the new decks is specified on the options cards. There are three recoding options. One allows the changing of card codes 0 through 15 to codes 0 through 9 on the new card. This option is the easiest to use and is sufficient for most recoding tasks. Another option replaces a new code for any code in a series of up to thirty old codes. The third recoding option allows the user to instruct the program to put a new code on the new card if the old code falls between user specified upper and lower limits. Other options enable the reorganization and addition of data. A variable generation option allows a list of user-specified old card columns to be added together to form a new summary variable. This option is designed to be used in conjunction with the item analysis program to create indices. A data transfer option enables the user to copy columns from any of the old decks to the new deck without modification. An addition option permits the user to add a constant to codes in a specified column and transfer it to the new deck. A copy option allows the punching of alphanumeric information into specified columns of each new card, an operation similar to gangpunching on the IBM 514 reproducer. Available also is a sequencing option which punches sequential numbers into specified columns of each new card. And finally, there is a blanking option which allows the user to blank out specific columns of his new card. In short, the recoding program is a small, card oriented, data management system which allows the user to recode and rearrange the data on his cards without programming knowledge.

There are two data analysis programs. One is a frequency distribution program which creates frequency distributions from integer variables ranging from 0 to 99. Zeros are counted but treated as nonresponses. For each response category in a variable, the number, percentage and cumulative percentage is printed. For each variable, the mode, median, mean and standard deviation is printed. This program is extremely simple to set up and requires only one control card and variable identification cards.

The second data analysis program is a cross-tabulation program. The cross-tabulation program creates tables of integer variables from 1 to 15. Zeros are treated as nonresponses and eliminated from statistical analysis. Cross-tabulation may be controlled by a third variable. For each table, the statistics chi square, contingency coefficient, phi coefficient, gamma and Z for gamma are printed. Percent by row and column are computed and

printed optionally. If third variable control is opted, at the end of each set of controlled tables, a partial gamma is provided. Tables may be generated by running all variables against each other, or by selecting variables to be run against the entire variable list. Also tables may be specified through the use of custom table cards which permit the user to specify row, column, and control variables and to merge code categories if desired.

Use And Acceptance Of SOLIB

In order to evaluate the usage of the SOLIB programs, a monitor was built into the library which counts each time a particular program is called. Users' Manuals were first distributed on February 11, 1972. Between the distribution of the manuals and December 24, 1972, it is estimated that the SOLIB programs were called 342 times. This estimate is probably low because there were times when the disk file which is used to store usage totals was destroyed in system crashes. One system crash lost the totals accumulated over nearly a two month period. In interpreting the calls, one should recognize that a call does not necessarily represent successful usage. The call of a program does not guarantee its running to a successful conclusion; deck set up errors, illegal codes and operator intervention could prevent a job from running satisfactorily. The table below lists the number of calls for each program as they were recorded.

TABLE 1. SOLIB Calls by Program,
February 11, 1972, to December 24, 1972

Program	Calls	Percent
Sample Number Checking	68	19.9
Wild Code Checkg	43	12.6
Listing	16	4.7
Item Analysis	13	3.8
Recoding	86	25.1
Frequency Distributions	60	17.5
Cross-tabulations	56	16.4
Total	342	100.0

The most popular program was the recoding program. Users have reported that the recoding program has not only been useful for the reorganization of data decks for further processing with the SOLIB programs, but for use in conjunction with other Computer Center programs as well. The sample number checking program received 19.9% of the calls. It has proved to be most helpful in editing data decks so that there are not duplicate or missing cards. This is essential where the amount of data requires two or more cards and the decks have to be arranged into a merged sequential file. The wild code checking program did not receive as much use as the sample number checking program. The item analysis program and the listing program only received 3.8% and 4.6% of the calls respectively. The analysis programs received considerably more use. The frequency distribution program received 17.5% of the calls while the cross-tabulation program received 16.4% of the calls.

The users of the SOLIB package may be divided roughly into three types: students using SOLIB as part of class assignments, students using SOLIB in conjunction with independent research projects, and faculty members using SOLIB for professional research.

In a sociology research methods course, students were required to use the SOLIB programs for the analysis of data they had collected as part of class assignments. At the outset, most were terrified of using the computer which appeared to them as a mysterious black box. However, the ease of set up for the data cleaning programs brought early success experiences which helped the students overcome their original fears. Most of the students were able to use the computer as a tool to gain control over their data in order to perform rather sophisticated analyses. Some students, after their initial success with the SOLIB package, recognized its limitations and continued their analysis with other Computer Center programs.

Several students have successfully used the SOLIB programs to analyze data collected for individual study projects. A few students, who have had no previous experience with computers or data processing, have, with the help of an instructor, been able to rapidly learn the logic of the SOLIB programs and process their data within a week's time. Students

who have collected data for analysis have been helped twice by the SOLIB programs. First, the programs save them from an enormous amount of tedious work; and secondly, the programs have a logical progression which guides them through the data analysis phase of the research process.

Several faculty members have also found the SOLIB programs to be helpful. These have included faculty members from the Speech, Political Science, Anthropology, History, Sociology, Physical Education, Health Education, Educational Research, and Psychology departments. Faculty members find the frequency distribution and recoding programs most useful. The frequency distribution program gives them a quick initial review of their data, and the recoding program helps them prepare their data for processing with other Computer Center programs.

To facilitate dissemination of information about the SOLIB programs, the author held a series of "SOLIB Seminars" during the early part of the Fall 1972 semester on the Brockport campus. Announcements of the seminars were sent to each department in which there might be persons who would benefit from the SOLIB programs. The seminars were held on four successive Wednesday evenings. Students as well as faculty members were invited to attend; the announcement stressed that no prior knowledge of computer programming was needed.

The first seminar was attended by 19 persons. During the next three meetigs, the attendance dwindled to 8 persons. The seminars covered the general logic of coding data from information collection schedules, the preparation of data cards, data cleaning problems, and of course, the use of the SOLIB programs. Participants were given decks of "dirty" data which they cleaned and analyzed as homework assignments. Those who faithfully attended the seminars were able to master the SOLIB package in a month's time. It is the author's impression that those who dropped out did so because of a lack of understanding of what they were doing rather than how they were doing it. While those who did not complete the seminar wanted to learn to use the computer, they did not understand the logic of research. Consequently, what they were doing made little sense and there was little motivation to continue. In short, understanding the rationale of the SOLIB package requires a minimal understanding of research methods. Future user's seminars will place a greater emphasis on the logic of the research process. The computer can be of little help if one does not know what he is doing methodologically.

Transportability And Availability

The value of SOLIB to potential users on campuses other than Brockport is dependent upon the ease with which it can be adapted for use on computers other than the 16K IBM 1130. As previously indicated, an 8K IBM 1130 version is currently operational which supports most of the features supplied in the 16K version. Transportability to campuses with other computers would require program modifications which would make the existing programs compatible with other FORTRAN compilers. To aid in the evaluation of the potential difficulty in converting the SOLIB programs, the table below indicates the departures from the USA Standard, Basic FORTRAN that are used in the SOLIB package.

TABLE 2. Features Used in the SOLIB Package not Supported by USA Standard, Basic FORTRAN

1. the $ and ' characters
2. three dimensional arrays
3. blank, 0, and 1 character carriage control
4. type statements with dimensions
5. the A and literal format types
6. mixed mode arithmetic
7. disk statements

Since the SOLIB programs do not use logical IF statements or variable names with more than five characters, conversion problems are minimal.

The author received substantial support from the National Science Foundation to develop the SOLIB programs and desires the programs and their documentation to become part of the public domain. Anyone wanting copies of the programs or the users' manuals may obtain them for the cost of reproduction from the author.